



Our Docket No.: 42P10195

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

Unice )  
Application No.: 10/037,530 ) Examiner: Mitchell, Jason D.  
Filed: January 3, 2002 ) Art Group: 2193  
For: Method and Computer Program Product )  
for Providing a Device Driver )

Commissioner of Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**DECLARATION UNDER 37 CFR 1.131 IN SUPPORT OF PRIOR INVENTION**

Sir:

We declare:

1. Intel Corporation is the assignee of the claims of the above captioned patent application ("the Application") and of the subject matter described therein.
2. Prior to November 29, 2001, the filing date of U.S. Patent Application Publication No. 2003/0101290 cited in a final Office Action mailed June 6, 2005, the invention claimed in the Application had been conceived and reduced to practice in the United States.
3. Attached Exhibit A is a redacted copy of an invention disclosure form describing the design of The Enhanced Special Programming Mode, and establishes that the subject matter claimed in the Application had been reduced to practice in the United States prior to November 29, 2001. Exhibit A (the invention disclosure) Declaration Under Rule 1.131 in Support of Application Application No. 10/037,530

describes “a solution for using a dynamically loading Linux kernel module across several versions of Linux. This means that a single binary complied module can load on any given version of Linux.” (Exhibit A, paragraph 1, lines 1-2). Exhibit A further describes “ship[ing] a driver without shipping the source code for that driver while still enabling the driver to run on any version of the Linux kernel.” This “enables the developer to continue to use the current driver binary on newer released kernels. As long as the API parameters match the newer kernel, the older version independent driver can load and run.” (Exhibit A, paragraph 3, lines 2-6).

4. Therefore, Exhibit A establishes that the subject matter claimed in the Application had been reduced to practice in the United States prior to November 29, 2001.

Furthermore, all statements made herein of our own knowledge are true and all statements made on information and belief are believed to be true, and further that these statements are made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both under section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application of any patent issuing thereon.

Dated: September 21, 2005

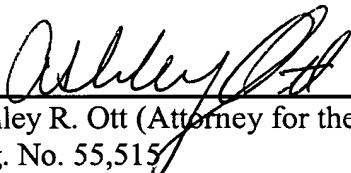
  
\_\_\_\_\_  
Ashley R. Ott (Attorney for the Assignee)  
Reg. No. 55,515

EXHIBIT A

INTEL CONFIDENTIAL

This patent application describes a solution for using a dynamically loading Linux kernel module across several versions of Linux. This means that a single binary compiled module can load on any given version of Linux. This binary module can also load on uni-processor and multi-processor versions of Linux.

Linux drivers, in order to function, must run using the exported kernel APIs. The exported kernel API names are specific to each version of the Linux kernel and to whether or not the kernel is enabled to support symmetric multiprocessing. This means that the driver source must be compiled for a specific version of the Linux kernel, if the driver is to work on a platform running a particular version of Linux. The common practice is to release a new compiled version of a driver for each major release of the Linux Kernel. Due to the frequency of the Linux kernel releases, this can be problematic for developers.

My invention enables a binary version of a driver to be shipped with a make file that can build and install the binary driver into the Linux kernel. This enables us to ship a driver without shipping the source code for that driver while still enabling the driver to run on any version of the Linux kernel. This also enables the single binary to run on platforms with one or more than one processor. This invention enables the developer to continue to use the current driver binary on newer released kernels. As long as the API parameters match in the newer kernel, the older version independent driver can load and run.

The method for detecting patent infringement is the following. When this method is used an address is required to be passed to the driver to enable the driver to begin importing symbols. This address is obtained by parsing "/proc/ksyms" for a specific label. The address is then passed down to the driver and the importing code uses this address to find the symbols for import. Anyone using this same method would need to pass this address in on the driver command line.

A programming model must be followed to allow the binary to be shipped, but as long as the model is followed the binary becomes portable across several versions of the Linux kernel.

There may be other methods for loading a driver on a version of Linux that the driver was not compiled on. The methods will differ in their flexibility and completeness of the solution. While researching prior art our solution may differ significantly from previous solutions.

